C语言课程设计

三层电梯状态机仿真程序

机电工程学院

16040310012

吕 铮

一．三层电梯状态机图

floor = IdleWhatFloorToGoTo(&up);

if (floor > 0 && up)

{

SetMotorPower(1);

\*state = MovingUp;

}

if (GetCloseDoorLight())

{

SetCloseDoorLight(false);

return;

}

floor = IdleWhatFloorToGoTo(&up);

if (floor > 0 && !up)

{

SetMotorPower(-1);

\*state = MovingDown;

}

if (GetOpenDoorLight())

SetOpenDoorLight(false);

if (GetCloseDoorLight())

SetCloseDoorLight(false);

if (GetCallLight(GetNearestFloor(), true))

{

SetDoor(GetNearestFloor(), true);

SetCallLight(GetNearestFloor(), true, false);

\*state = DoorOpen;

}

if (GetCallLight(GetNearestFloor(), false))

{

SetDoor(GetNearestFloor(), true);

SetCallLight(GetNearestFloor(),false, false);

\*state = DoorOpen;

}

二．流程图

外部 内部

呼叫 指令 2s

延

时

标

**Door Closing**

目

达 层

到 楼

三．电梯功能测试

1. 当按下门外up/down按键后，到达该楼层，并且箱门打开，无障碍下2s后箱门关闭，按键灯熄灭。
2. 当电梯停在2或3楼且处于空闲状态10s后，自动下降到1楼。
3. 当箱门处于开门状态，并有障碍物时，通过红外感应箱门一直处于打开状态，直至障碍物消失。
4. 当按下门内1，2或3楼按键后，电梯上升至相应楼层，箱门打开，无障碍2s后关闭，同时门内按键灯熄灭。
5. 电梯在静止状态下按下门内开门/关门键，箱门打开/关闭，在运动状态下不执行命令，同时按键灯熄灭。
6. 当电梯经过2楼时正好有呼叫指令，则电梯停下，箱门打开后关闭，在通往目标楼层。

四．部分状态机代码

bool up;

void StateIdle(int \*state)

{

int floor; bool up;

floor = IdleWhatFloorToGoTo(&up); // Event

if (floor > 0 && up)

{

SetMotorPower(1);

\*state = MovingUp; // 进入MovingUp状态

}

floor = IdleWhatFloorToGoTo(&up);

if (floor > 0 && !up)

{

SetMotorPower(-1);

\*state = MovingDown; // 进入MovingDown状态

}

if (GetOpenDoorLight()) //门内

{

SetDoor(GetNearestFloor(), true); //开门

SetOpenDoorLight(false); //消费开门键

\*state = DoorOpen; //进入DoorOpen状态

}

if (GetCallLight(GetNearestFloor(), true)) //门外

{

SetDoor(GetNearestFloor(), true); //开门

SetCallLight(GetNearestFloor(), true, false); //关灯

\*state = DoorOpen; //进入DoorOpen状态

}

if (GetCallLight(GetNearestFloor(), false)) //门外

{

SetDoor(GetNearestFloor(), true); //开门

SetCallLight(GetNearestFloor(),false, false); //关灯

\*state = DoorOpen; //进入DoorOpen状态

}

if (GetCloseDoorLight())

{

SetCloseDoorLight(false); //消费按键

return;

}

}

void StateMovingUp(int \*state)

{

int floor;

floor = GoingUpToFloor();

if (fabs(GetFloor() - floor) < Lib\_FloorTolerance)

{

SetMotorPower(0); //停止

SetDoor(GetNearestFloor(), true); //开门

\*state = DoorOpen; //进入DoorOpen状态

if (GetNearestFloor() != Lib\_FloorNum)

SetCallLight(GetNearestFloor(), true, false); //消费门外up键

else (SetCallLight(GetNearestFloor(), false, false)); //消费顶层down键

}

SetPanelFloorLight(GetNearestFloor(),false); //消费门内该楼层按键

if (GetOpenDoorLight())

SetOpenDoorLight(false); //消费门内开门灯

if (GetCloseDoorLight())

SetCloseDoorLight(false); //消费门内关门灯

}

void StateMovingDown(int \*state)

{

int floor;

floor = GoingDownToFloor();

if (fabs(GetFloor() - floor) < Lib\_FloorTolerance)

{

SetMotorPower(0); //停止

SetDoor(GetNearestFloor(), true); //开门

\*state = DoorOpen; //进入DoorOpen状态

if (GetNearestFloor() != 1)

SetCallLight(GetNearestFloor(), false, false); //消费门外up键

else SetCallLight(1, true, false); //消费1层up键

}

SetPanelFloorLight(GetNearestFloor(), false); //消费门内该楼层按键

if (GetOpenDoorLight())

SetOpenDoorLight(false); //消费门内开门灯

if (GetCloseDoorLight())

SetCloseDoorLight(false); //消费门内关门灯

}

void StateDoorOpen(int \*state)

{

int floor;

floor = GetNearestFloor(); //获取当前楼层

if (GetCloseDoorLight())

{

SetDoor(floor,false); //关门

SetCloseDoorLight(false); //消费门内关门灯

\*state = DoorClosing; //进入DoorClosing状态

}

if (IsDoorOpen(floor))

{

SetDoor(floor,false); //自动关门

\*state = DoorClosing; //进入DoorClosing状态

}

if (GetOpenDoorLight())

SetOpenDoorLight(false); //消费门内开门灯

}

void StateDoorClosing(int \*state)

{

int floor;

floor = GetNearestFloor(); //获取当前楼层

if (GetOpenDoorLight())

{

SetDoor(floor, true); //开门

SetOpenDoorLight(false); //消费门内开门灯

\*state = DoorOpen; //进入DoorOpen状态

}

if (GetCloseDoorLight())

SetCloseDoorLight(false); //消费门内关门灯

if (IsBeamBroken()) //红外探测

{

SetDoor(floor, true); //自动开门

\*state = DoorOpen; //进入DoorOpen状态

}

if (IsDoorClosed(floor))

\*state = Idle; //进入Idle状态

}

五．测试效果图







